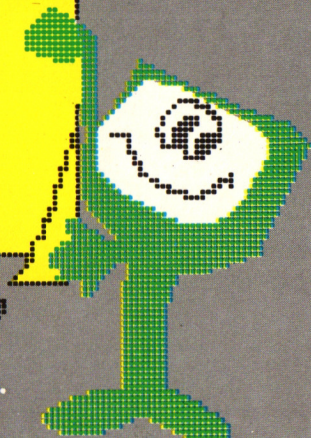
20 VIDEOLEZIONI DI BASIC
PER IMPARARE CON L'MSX

VIDEOBASIC MSX
Pubblicazione quattordicinale
edita dal Gruppo Editoriale Jackson
Direttore Responsabile:
Giampietro Zanga
Direttore e Coordinatore
Editoriale: Roberto Pancaldi
Autore: Softidea -
Via Indipendenza 88-90 - Como
Redazione software:
Michele Casartelli
Francesco Franceschini
Progetto grafico:
Studio Nuovidea - via Longhi, 16 - Milano
Impaginazione:
Moreno Confalone
Illustrazioni:
Cinzia Ferrari, Silvano Scolari
Fotografie:
Marcello Longhini
Distribuzione: SODIP
Via Zuretti, 12 - Milano
Fotocomposizione: Lineacomp S.r.l.
Via Rosellini, 12 - Milano
Stampa: Grafika '78
Via Trieste, 20 - Pioltello (MI)
Direzione e Redazione:
Via Rosellini, 12 - 20124 Milano
Tel. 02/6880951/5

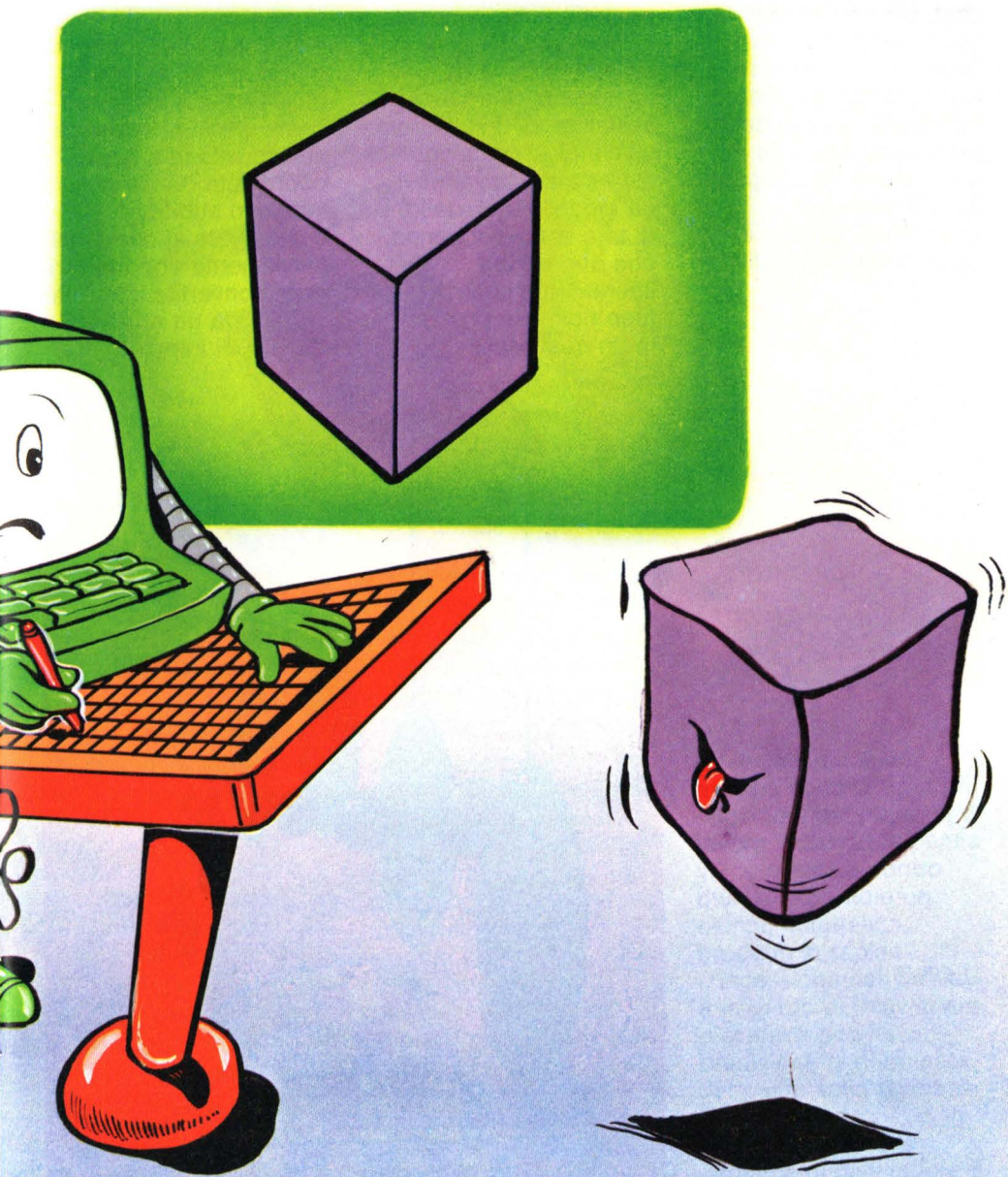
Autorizzazione alla pubblicazione Tribunale di Milano n° 422 del 22-9-1984
Spedizione in abbonamento postale Gruppo II/70 (autorizzazione della Direzione Provinciale delle PPTT di Milano).
Prezzo del fascicolo L. 8.000
Abbonamento comprensivo di 5 raccoglitori L. 165.000
I versamenti vanno indirizzati a: Gruppo Editoriale Jackson S.p.A. - Via Rosellini, 12 20124 Milano, mediante emissione di assegno bancario o cartolina vaglia oppure utilizzando il c.c.p. n° 11666203.
I numeri arretrati possono essere richiesti direttamente all'editore inviando L. 10.000 cdu. mediante assegno bancario o vaglia postale o francobolli.
Non vengono effettuate spedizioni contrassegno.

SOMMARIO

Introduzione

L'elaborazione delle immagini è uno degli aspetti più spettacolari ed affascinanti di un computer. Tutti abbiamo presente le immagini trasmesse dalla televisione di sigle e disegni sviluppati con un calcolatore grafico o di progetti ed esplosi di automobili ottenuti con sistemi elettronici dedicati. Sono processi che necessitano di molta memoria e di velocità di elaborazione elevatissime e quindi quasi assenti nelle macchine di qualche anno fà.
Non oggi, però. Lo dimostra il fatto che la lotta commerciale tra le case produttrici si svolga anche "a colpi" di risoluzione grafica, di pixel indirizzabili di software più o meno capace di produrre, gestire, animare e muovere le immagini.
Entriamo in questo nuovo ambiente esaminando alcune periferiche dedicate: la penna ottica e la tavoletta grafica.

La tavoletta grafica

Sin dall'antichità il disegno si è posto come uno dei mezzi di espressione e comunicazione tra i più utili, significativi e immediati.
Era chiaro che il computer, non poteva trascurare la grafica.
Dopo i primi passi, timidi ed incerti, negli anni passati, i possibili utilizzi degli elaboratori elettronici in questo settore stanno infatti diventando sempre più numerosi ed importanti, tanto che non sembra difficile pronosticare un futuro, non troppo lontano, fatto di immagini ed animazioni totalmente gestite o composte con l'ausilio del calcolatore.
I primi esempi sono già nei cinematografi.
Si parla poi (ma non è fantascienza) della possibilità di realizzare nuovi film con interpreti scomparsi da anni e "simulati da computer".
Ovviamente, in questi casi si tratta di computer da milioni di dollari; tutti i personal computer comunque hanno notevoli capacità grafiche che li rendono utili - per non dire indispensabili - in un ampio campo di applicazioni: dalla progettazione all'urbanistica, dalla medicina agli affari, dalla didattica ai giochi. Pur essendo una macchina essenzialmente numerica, il computer è quindi in grado di comporre disegni e grafici. Naturalmente, affinché le immagini

HARDWARE

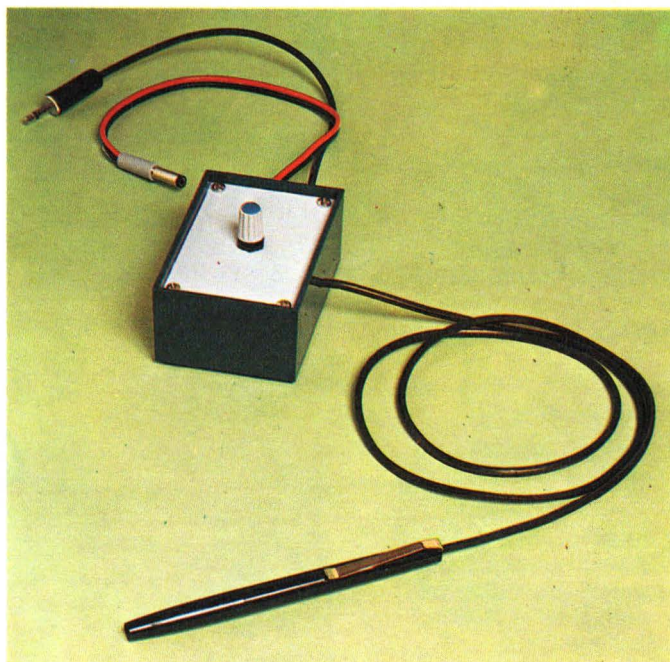
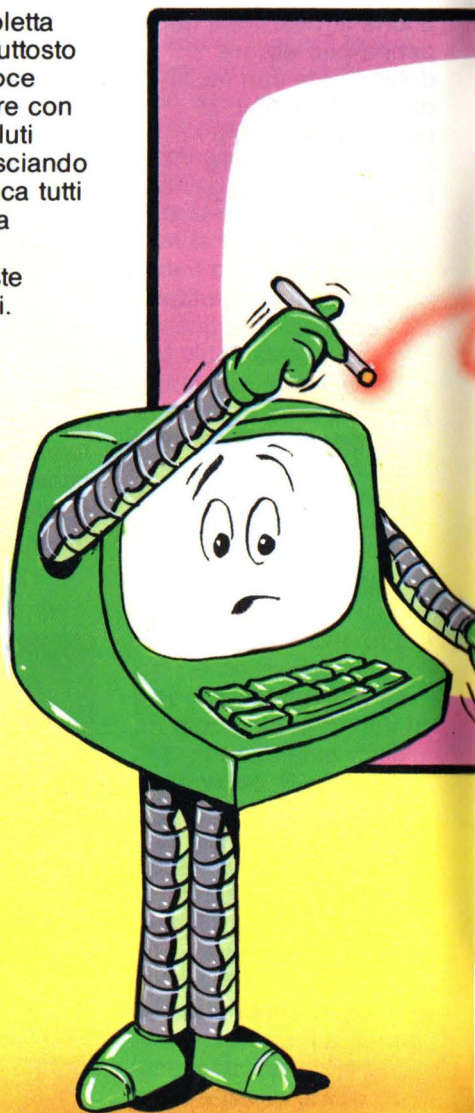
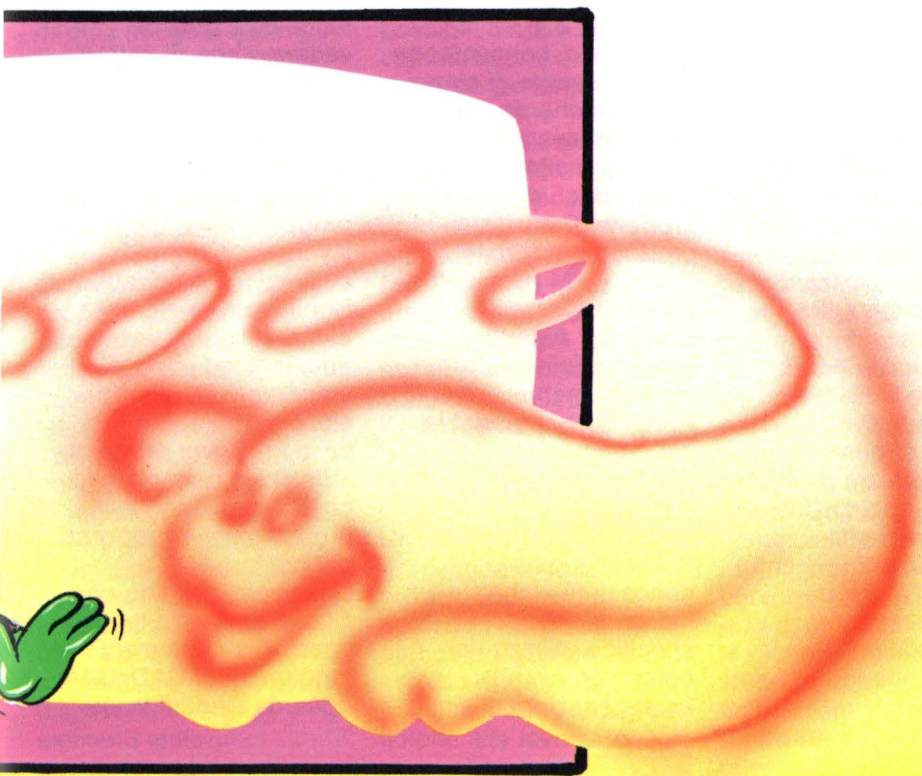
possano essere trattate, sono necessarie due cose: una codifica delle immagini stesse sotto forma di numeri e delle periferiche in grado di comunicare al computer queste immagini.
La tavoletta grafica è appunto un dispositivo periferico che consente di realizzare disegni in modo semplice ed immediato. Si tratta, in sostanza, di una piccola tavola (di dimensioni estremamente variabili da modello a modello) e di una specie di penna, che può essere liberamente posizionata (con notevole precisione) in un qualsiasi punto di questa tavola.
Una volta selezionato il punto che si desidera includere nel disegno basta premere la penna sulla tavoletta e le coordinate del punto vengono subito comunicate al computer (ovviamente dopo essere state convertite in forma digitale da un apposito circuito di interfaccia). Il

complesso delle coordinate, che vengono interessate durante il movimento della penna, compone via via il disegno, che appare così sullo schermo proprio come accadrebbe se si passasse una matita sopra un comune foglio di carta.
Il principio di funzionamento che consente di eseguire queste operazioni è normalmente basato su un fitto reticolo di fili incrociati, disposto appena sotto la superficie del piano di lavoro. Quando la penna viene premuta contro questa superficie un campo magnetico, prodotto in rapida successione dai vari fili, è in grado di individuare e stabilire con precisione le coodinate del punto sulla tavoletta.
È ovvio che la risoluzione (cioè la capacità di distinguere il più piccolo movimento della penna) dipende dal

numero di fili che compongono il reticolo: nei modelli per uso hobbystico essa è logicamente molto più limitata rispetto agli strumenti professionali, arrivando comunque, anche nei modelli più economici, alle 10 linee per millimetro (la tavoletta riesce cioè ad avvertire variazioni nella posizione della penna fino a valori superiori di 1/10 di millimetro).
Come qualunque altra periferica, la tavoletta grafica serve comunque soltanto per comunicare informazioni al computer, o meglio al programma, che decide come utilizzare le informazioni ricevute.
Insieme alla tavoletta deve quindi essere utilizzato anche un programma che converta i comandi impartiti con la penna in precise istruzioni eseguibili dal computer. Per i personal più diffusi questi programmi vengono forniti assieme alla tavoletta, e costituiscono molto spesso parte integrante di tutto il corredo necessario per disegnare con il calcolatore.

La penna ottica

Un altro dispositivo che consente di disegnare in modo più economico e meno ingombrante della tavoletta grafica è la cosiddetta penna ottica (o penna luminosa).
Si tratta di un sensore, di forma molto simile ad una comune penna, in grado di rivelare le variazioni di luminosità alle quali viene sottoposto. Il suo impiego è molto simile a quello della penna di una tavoletta grafica, con la sola differenza che il piano di lavoro è costituito - anziché da un supporto esterno - dal video stesso del computer. Infatti, è sufficiente appoggiare la penna in un qualsiasi punto del video ed il computer riceve (come al solito convertite in forma digitale) le coordinate del punto selezionato.
Per capire il modo in cui ciò possa avvenire bisogna richiamarsi per un momento al principio di funzionamento delle unità video. Ricordi? Un raggio (o fascio) di elettroni scandisce in continuazione uno strato di sostanze sensibili alla luce (i fosfori), poste sulla superficie interna dello schermo, provocandone o meno la luminescenza. La penna luminosa è proprio un dispositivo in grado di distinguere i punti illuminati da quelli non illuminati da tale raggio. Dato che la posizione del fascio elettronico è nota in ogni istante,

anche la posizione della penna può essere determinata con facilità dai circuiti adibiti al controllo del video.
La precisione della penna ottica risulta comunque notevolmente più limitata di una tavoletta grafica, non potendo, in nessun caso, superare la definizione propria dello schermo. Inoltre essa costringe a lavorare con il braccio alzato e con gli occhi sempre fissi sul video. Tuttavia, non tutte le penne ottiche sono uguali. Differenze più o meno piccole nelle dimensioni e nella luminosità richiesta allo schermo video ne rendono infatti alcune migliori delle altre. È naturale che le migliori siano quelle con le dimensioni più piccole e che richiedono una minore luminosità allo schermo, visto che affaticano di meno sia il braccio che la vista. Ancora una volta, comunque, risultano indispensabili per il funzionamento di questo dispositivo i circuiti elettronici di interfaccia ed il programma di utilizzo impiegati, per rendere compatibili al calcolatore le informazioni ricevute

dall'esterno.
Ad ogni modo, per lavori di una certa complessità e precisione la penna ottica non può certo essere paragonata o sostituita alla tavoletta grafica. Essa è piuttosto un comodo e veloce mezzo per indicare con rapidità i punti voluti sullo schermo, lasciando alla tavoletta grafica tutti i lavori nei quali la complessità e la precisione richieste risultano superiori.

HARDWARE

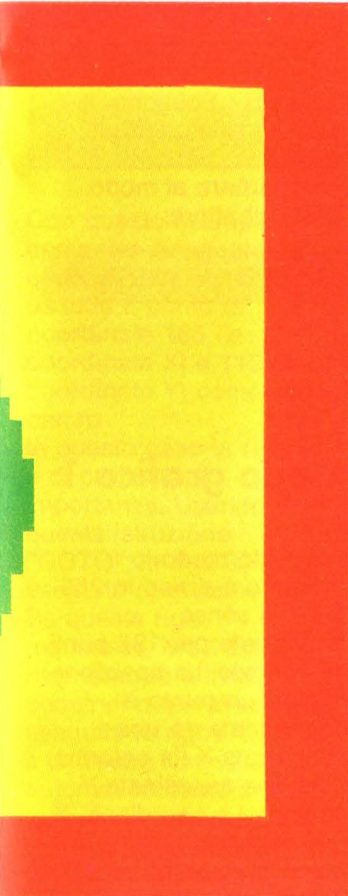
CAD/CAM

Abbiamo visto che l'introduzione dei computer (ed in particolar modo dei personal) propone, oltre ai tradizionali utilizzi, anche la possibilità di produrre disegni tramite elaboratore.
Tale possibilità non è comunque limitata alle quotidiane attività di ufficio od al puro divertimento: tra le molteplici applicazioni, particolare risalto meritano infatti i recenti utilizzi della grafica computerizzata nel campo della progettazione industriale.
Questi utilizzi, inizialmente nati grazie agli studi fatti nelle grosse industrie automobilistiche, hanno visto il computer collaborare ed affiancarsi in misura sempre maggiore ai progettisti e ai disegnatori, consentendo il superamento di alcuni problemi che da sempre condizionavano in maniera molto pesante la fase di studio e progetto di qualsiasi prodotto industriale.
CAD e CAM (Computer Aided Design e Computer Aided Manufacturing, rispettivamente Progettazione assistita dal computer e Produzione assistita dal computer) sono proprio le abbreviazioni - universalmente conosciute - che indicano questo specifico utilizzo del computer.
Le applicazioni di simili tecnologie (poiché vedremo che di vere e proprie tecnologie si tratta) sono ormai numerosissime ed estremamente diffuse: automatizzare progettazione, lavorazione e costruzione ha difatti significato - e significa tuttora - una notevole riduzione ed ottimizzazione degli sforzi e soprattutto dei costi in diversi processi industriali, permettendo così la creazione di prodotti migliori a prezzi più contenuti.
L'esempio classico che illustra al meglio le possibilità attualmente raggiunte dal CAD/CAM riguarda l'industria dei

La TRACK-BALL è una periferica di Input estremamente valida per produrre grafici in alta risoluzione.

circuiti integrati (o chip). Sai già che un circuito integrato non è altro che un microscopico circuito elettrico posto in una piastrina di silicio. Sembrerebbe quindi cosa facile poter rimpicciolire direttamente il circuito di progetto (realizzato necessariamente a "misura d'uomo" e quindi con dimensioni tali da potervi lavorare sopra ad occhio nudo) nel prodotto finito, cioè in un microscopico chip. Niente di più errato. Sin dalla fase di progetto occorre infatti tener presente che bisogna ottimizzare e minimizzare sia gli spazi disponibili che gli elementi elettronici costituenti il circuito, così che alla fine risulti occupata la minor superficie possibile. Questo lavoro implica per forza di cose migliaia di calcoli con i quali esaminare tutte le possibili disposizioni e combinazioni, e che risultano pertanto assolutamente impossibili da affrontare senza l'ausilio di un computer appositamente predisposto.
Sul video del progettista l'elaboratore - grazie al programma CAD - visualizza allora le possibili disposizioni dei componenti elettronici, limitando (od addirittura evitando) la lunga e noiosa fase di disegno manuale dei vari circuiti.
Una volta risolto il progetto, anche la produzione dei chip richiede l'adozione di particolari cautele ed il rispetto di determinate tolleranze alle macchine adibite alla lavorazione, cose queste che soltanto un computer programmato per il CAM è in grado di coordinare con affidabilità e sicurezza.
Alla fine del processo produttivo la tecnologia CAD/CAM è quindi in grado di offrire dei circuiti integrati con qualità e prestazioni nettamente superiori a quelle altrimenti ottenibili e soprattutto ad un prezzo di gran lunga più conveniente.
Quello che abbiamo fatto era solo un esempio: invece di chip potevano essere telai per automobile o componenti di motori a reazione, ed il discorso sarebbe stato praticamente identico.
È chiaro che l'uso di una simile tecnica per la progettazione e la produzione richiede computer di grandi dimensioni ed elevata potenza di calcolo (e pertanto di esclusiva portata delle grosse industrie). Già adesso stanno comunque cominciando ad apparire i primi programmi per personal computer: nel giro di pochi anni anche le piccole industrie o gli studi professionali dovrebbero quindi avere a propria disposizione questi utili e potenti mezzi di progettazione e produzione, permettendo così di allargare ulteriormente il già vasto repertorio di prodotti attualmente fabbricati utilizzando il CAD/CAM.

Alta risoluzione

All'interno del tuo MSX esiste un circuito integrato chiamato VDP (Processore Video) la cui funzione è quella di attingere alle informazioni contenute nella memoria video, ed utilizzarle per creare dei segnali, che, opportunamente codificati, possano raggiungere il televisore od il monitor, generando delle immagini.
Il VDP è un dispositivo programmabile, cioè è in grado di funzionare in più modi, a seconda dei comandi che gli giungono in ingresso.
Il primo di questi è il cosiddetto "modo": è quello disponibile al momento dell'accensione ed è anche il più semplice da usare.

Modo testo 1

In questo modo è possibile inserire sullo schermo un massimo di 40 caratteri per riga e lo schermo dispone di 24 di tali righe. Ogni posizione sullo schermo è definita dalla coordinata X (che indica la colonna) e dalla coordinata Y (che indica la riga). La coordinata X più piccola è 0 e corrisponde alla colonna più a sinistra, mentre quella più elevata è 39, che corrisponde alla colonna all'estrema destra. Le coordinate Y vanno da 0

(corrispondente alla riga superiore) fino a 23 (corrispodente alla riga inferiore).
Con l'istruzione "WIDTH" è possibile indicare al computer il numero di caratteri desiderati per riga.

Con l'istruzione "LOCATE" il cursore può essere spostato in qualsiasi posizione desiderata sullo schermo.
Questo modo testo (modo testo 1) può essere attivato con l'istruzione "SCREEN 0".
Esempio:

```
10 SCREEN 0
20 WIDTH 40
30 LOCATE 15,10 : PRINT "CIAO"
```

In questo esempio nella riga 10 viene attivato il modo testo 1. La riga 20 dà la possibilità di inserire 40 caratteri in una riga.
Nella riga 30 il cursore è spostato alla posizione 15 (16-esima colonna) della riga 10 (11-esima riga) dove viene scritta la parola "CIAO".
Quando MSX-BASIC è a livello di comandi oppure deve elaborare un'istruzione "INPUT", lo schermo è sempre nel modo testo.
Questo modo non consente l'utilizzo degli sprite.

Modo testo 2

Questo modo testo consente di avere sullo schermo un massimo di 32 caratteri per riga. Anche in questo caso ogni posizione sullo schermo è definita dalla coordinata X per la colonna e dalla coordinata Y per la riga.

La coordinata X è un numero compreso tra 0 per la colonna di sinistra e 31 per la colonna di destra. La coordinata Y va da 0 per la riga superiore a 23 per la riga inferiore.
Il numero massimo di caratteri per riga può essere determinato usando l'istruzione "WIDTH".
Usando l'istruzione "LOCATE" il cursore può essere spostato in qualsiasi posizione dello schermo.
Questo modo testo (modo testo 2) è attivato con l'istruzione "SCREEN 1". Ecco un esempio:

```
10 SCREEN 1
20 WIDTH 32
30 LOCATE 15,10 : PRINT "CIAO"
```

Questo esempio da esattamente lo stesso risultato del precedente, che descriveva il modo testo 1.
Nella riga 10, è stato attivato il modo testo 2.
Nella riga 20 si è creata la possibilità di usare 32 caratteri per riga.
Ricordati che nel modo testo 2 ci sono aree destinate al bordo blu-verde nella parte superiore e nella parte inferiore dello schermo e che è possibile cambiare il colore di queste aree usando l'istruzione "COLOR". Possono essere visualizzati gli sprite.
Per ritornare al modo testo 1, battere:

```
SCREEN 0 : WIDTH 40
```

Modo grafico 1

In questo modo lo schermo è diviso in 256 punti in senso orizzontale per 192 punti in verticale. La posizione di ciascun punto è determinata da una coordinata X (la colonna) e da una coordinata Y (la riga). La coordinata X può essere qualsiasi numero compreso tra 0 (per la colonna più a sinistra) e 255 (per la colonna più a destra). La coordinata Y può variare da 0 (per la riga superiore) a 191 (per la riga inferiore).
In questo modo, per creare immagini sullo schermo possono essere usate le istruzioni grafiche. Questo modo è attivato con l'istruzione "SCREEN 2".

Esempio:

```
10 SCREEN 2
20 CIRCLE (185,115),10
30 GOTO 30
```

Con questo esempio il computer disegnerà un cerchio sullo schermo, usando il punto di coordinate 185 (la coordinata X) e 115 (la coordinata Y) come suo centro.
In questo caso la riga 30 è di particolare importanza. Usando questa istruzione "GOTO", la riga 30 viene eseguita ripetutamente. Se questa riga mancasse, MSX-BASIC ritornerebbe a livello comandi dopo aver eseguito le righe 10 e 20 e verrebbe riattivato automaticamente il modo testo 1. Considerando la velocità con cui MSX-BASIC opera, non si avrebbe nemmeno il tempo di vedere il cerchio.

È possibile interrompere il programma premendo contemporaneamente i tasti CTRL e STOP.

Modo grafico 2

In questo modo lo schermo è diviso esattamente allo stesso modo del modo grafico 1.
La differenza tra i due è che le figure che compaiono sullo schermo sono tutte costituite da piccoli quadratini, che misurano ciascuna 4x4 punti.
L'immagine è per questo meno definita che in SCREEN, ma ha il vantaggio che ogni punto può assumere un colore diverso. Per tale motivo questo modo viene detto anche "MULTICOLOR".
Il modo grafico 2 è attivato con l'istruzione "SCREEN 3".
Lavorando in questo modo, possono essere usate le istruzioni grafiche.

Ecco ora un esempio:

```
10 SCREEN 3
20 LINE (125,10) - (200,85)
30 GOTO 30
```

Con questo programma verrà disegnata una linea dal punto di coordinate 125,10 (rispettivamente coordinata X e Y), fino al punto di coordinate 200,85 (rispettivamente coordinate X e Y).
La linea è costituita da tanti piccoli quadratini. È possibile interrompere il programma premendo simultaneamente i tasti CTRL e STOP.
Prova ora a vedere cosa succede quando cambi la riga 10 in:

```
10 SCREEN 2
```

Esegui di nuovo il programma.

Cancellazione dello schermo

È possibile cancellare o "ripulire" lo schermo con l'istruzione "CLS". La si può usare con tutti i modi dello schermo.

LINGUAGGIO

Le istruzioni grafiche

L'MSX-BASIC possiede un set di istruzioni grafiche completo e potente.

Naturalmente queste istruzioni funzionano solo se lo schermo si trova in uno dei modi grafici di codice 2 o 3.

PSET e PRESET

Queste sono le istruzioni grafiche più semplici, e consentono di accendere o spegnere un pixel sullo schermo grafico. Come abbiamo

già visto, un pixel è il più piccolo elemento controllabile di un disegno sullo schermo. Nel modo schermo 2, sono disponibili 256 x 192 pixel, ciascuno dei quali è individuato da una coppia di numeri detti "COORDINATE". Il primo dato da specificare quando si vuole accendere o spegnere un pixel sono proprio le sue coordinate.

L'MSX-BASIC richiede che le coordinate siano racchiuse tra parentesi tonde e separate tra loro da virgole, così:

```
PSET (10,20)
```

accende il pixel di coordinata X=10 e coordinata Y=20, cioè a 10 pixel dal margine sinistro e a 20 pixel dal margine superiore con il colore specificato dall'ultima istruzione COLOR per l'inchiostro.

```
PRESET (255,191)
```

spegne il pixel nell'angolo in basso a destra di coordinata X=255 e coordinata Y=192, cioè lo rende del colore specificato dall'ultima istruzione COLOR per lo sfondo.
Un altro modo di specificare le coordinate consiste nel fornire la distanza del punto richiesto dall'ultimo punto a cui ci si è riferiti. Per fare questo bisogna usare la parola chiave STEP prima delle coordinate, che verranno sommate algebricamente a quelle dell'ultimo punto individuato.
Le coordinate, che in questo caso si dicono "relative" potranno naturalmente essere sia positive che negative.

Esempi

```
PSET STEP (2,2)
```

Indica che si vuole accendere il pixel che si trova 2 pixel a destra e 2 pixel sotto l'ultimo pixel utilizzato da qualsiasi altra istruzione grafica eseguita in precedenza.

```
PRESET STEP (-1,0)
```

Spegni il pixel all'immediata sinistra dell'ultimo: se ad esempio questo aveva coordinate (20,20), l'istruzione si riferisce a quello di coordinate (19,20).

Con le istruzioni PSET e PRESET è possibile anche assegnare ai pixel colori diversi da quelli specificati nella precedente istruzione COLOR.
Per fare questo, basta specificare il codice del colore dopo le coordinate del pixel, separandolo con una virgola.
Usate in questo modo, le istruzioni PSET e PRESET risultano praticamente identiche.

```
PSET (128,96),15
```

Colora il pixel di coordinate 128,96 (al centro dello schermo) di bianco (codice 15).
Lo stesso risultato si ottiene usando la parola chiave PRESET.

Sintassi delle istruzioni

PSET [STEP] (X,Y) [,COLOR]
PRESET [STEP] (X,Y) [,COLOR]

POINT

Con la funzione POINT si può conoscere il codice del colore di un qualsiasi punto dello schermo grafico.
Le coordinate del punto vanno passate alla funzione nello stesso modo di PSET e PRESET, e il risultato fornito è il codice richiesto.
È ammesso l'uso di STEP.

```
CP = POINT (10,10)
```

Assegna alla variabile CP il colore del pixel di coordinate (10,10).

```
PSET (1,1), POINT (10,10)
```

rende il pixel di coordinate (1,1) uguale al pixel di coordinate (10,10) assegnandogli lo stesso colore.

Sintassi della funzione

POINT [STEP] (X,Y)

LINE

Questa istruzione serve a tracciare sullo schermo dei segmenti che uniscono 2 punti stabiliti.
Basta scrivere la parola LINE seguita dalle coordinate dei punti di partenza e di arrivo separate da un trattino (-), e il segmento verrà tracciato.
Il colore usato sarà quello specificato per l'inchiostro nell'ultima istruzione "COLOR" eseguita.

```
LINE (0,0) - (255,191)
```

Traccia il segmento che unisce il punto di coordinate (0,0) con il punto di coordinate (255,191) (la diagonale dello schermo).
Anche qui le coordinate possono essere espresse in modo relativo facendo uso di "STEP".
Le coordinate del punto di partenza, poi, si possono omettere nel caso in cui questo coincida con il punto riferito nell'ultima istruzione grafica eseguita.
Se si vuole usare un colore diverso da quello specificato per l'inchiostro, bisogna aggiungere il suo codice dopo le coordinate del punto in arrivo, separato da una virgola.

```
LINE (10,10) - STEP (10,0)
```

Disegna il segmento che unisce il punto di coordinate (10,10) con quello che si trova 10 pixel a destra di quest'ultimo.

```
LINE (0,0) - (5,5),15
```

Traccia il segmento che unisce i punti di coordinate (0,0) e (5,5) usando il colore bianco (codice 15).

```
LINE - STEP (0,20),1
```

Traccia un segmento verticale di 20 pixel sotto l'ultimo punto utilizzato da una istruzione grafica.
Con l'istruzione LINE è anche possibile tracciare dei rettangoli, vuoti o pieni di colore.

Per fare questo basta specificare le coordinate che si userebbero per tracciare la diagonale del rettangolo, e aggiungere, dopo il codice del colore, l'opzione ,B per avere il rettangolo vuoto e ,BF per quello pieno. (B sta per BOX e F per FILL).

```
LINE (10,10) - (15,20), 1, BF
```

Traccia un rettangolo pieno di colore nero (codice 1) con diagonale che unisce i punti (10,10) e (15,20).

Sintassi dell'istruzione

LINE [[STEP] (X1,Y1)] - [STEP] (X2,Y2) [,COL] [,B [F]]

CIRCLE

Questa istruzione serve a tracciare sullo schermo dei cerchi o degli archi di cerchio o di ellisse.
I primi dati da fornire all'istruzione sono le coordinate del centro, secondo la procedura solita per indicare i punti, e la misura del raggio espressa in pixel. Facoltativamente possono essere dati il colore, l'angolo di partenza e di arrivo nel caso di archi, e l'eccentricità per le ellissi.

```
CIRCLE (90,90),15
```

Traccia la circonferenza di centro (90,90) con raggio 15.

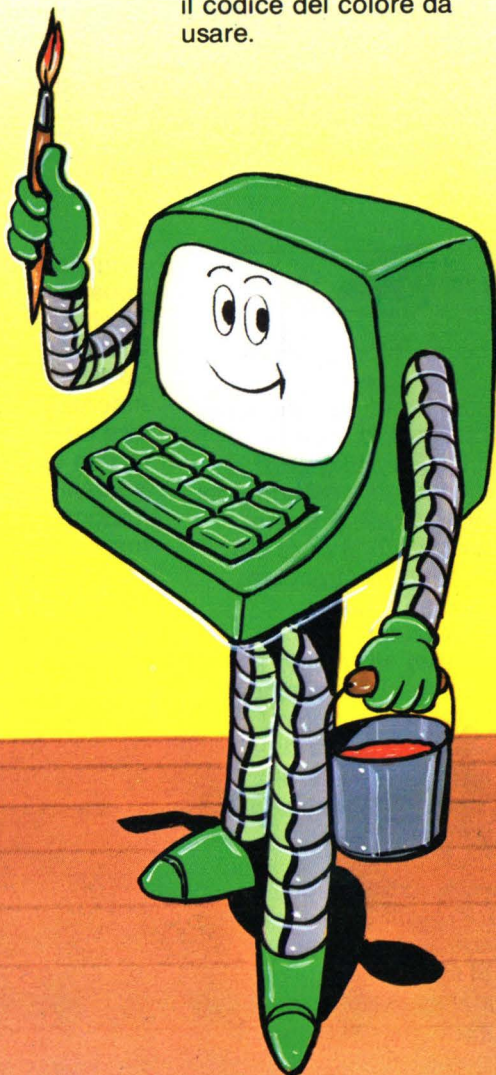
Sintassi dell'istruzione

CIRCLE [STEP] (X,Y), R, [,COL] [,ANG.IN] [,ANG.FINE] [,RO/RV]

PAINT

Questa istruzione serve a colorare aree di schermo delimitate da linee chiuse. Per fare ciò basta specificare, dopo la parola chiave PAINT, le coordinate di un punto interno alla linea e il codice del colore da usare.

L'istruzione, però, funziona solo se la linea che racchiude l'area da colorare è perfettamente chiusa ed è dello stesso colore usato per riempirla.

```
PAINT (13,24),15
```

Colora l'area racchiusa da una linea chiusa che contiene il punto di coordinate (13,24) con il colore bianco.

Se la linea non è chiusa o non è di colore bianco, si avranno delle fuoriuscite di colore, e tutto lo schermo ne sarà invaso.

Sintassi dell'istruzione

PAINT [STEP] (X,Y),COL [COL.BORDO]

DRAW

Le istruzioni grafiche viste fino ad ora sarebbero sufficienti per qualsiasi disegno.
Il BASIC MSX, però, mette a disposizione del programmatore uno strumento ancora più potente per tracciare disegni in alta risoluzione.
Questo strumento è costituito dal cosiddetto "Macrolinguaggio Grafico", o, per dirla all'inglese, Graphic Macro Language (GML).
Il macrolinguaggio grafico consiste in una serie di semplici istruzioni capaci di comandare una specie di matita immaginaria sullo schermo, facendole tracciare delle linee o semplicemente spostandola da un punto all'altro.
L'istruzione DRAW serve ad eseguire le macroistruzioni grafiche, che devono essere poste in una stringa.
Vediamo ora le più importanti macroistruzioni, cercando di capire meglio con degli esempi.

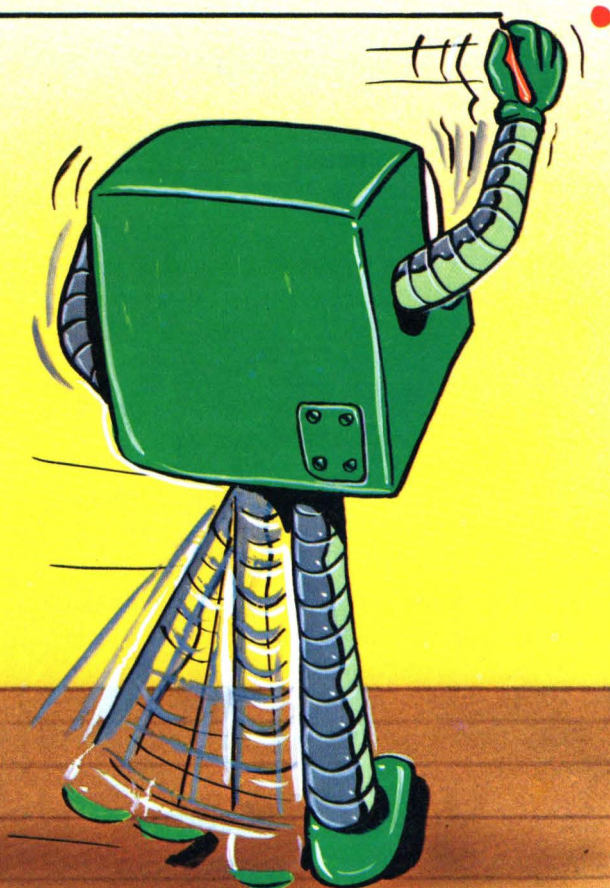
Le macro U, D, L, R

Queste istruzioni servono a spostare la nostra matita tracciando delle linee parallele ai bordi dello schermo.

U - muove verso l'alto (UP)

D - muove verso il basso (DOWN)

R - muove verso destra (RIGHT)

L - muove verso sinistra (LEFT).

Esempi

DRAW "R10"

traccia una linea spostando la matita 10 pixel a destra.
Il numero dopo la macroistruzione R indica la distanza da percorrere espressa in pixel.

DRAW "U20R20D20L20"

Traccia un quadrato di lato 20 pixel.
Come vedi, più macroistruzioni possono essere eseguite in una sola istruzione DRAW.

Le macro E, F, G, H

Queste istruzioni servono a spostare la nostra matita tracciando delle linee diagonali.

E - in alto a destra

F - in basso a destra

G - in basso a sinistra

H - in alto a sinistra.

Esempi

DRAW "E10F10G10H10"

traccia un rombo di lato 10.

Le macro B ed N

Queste macroistruzioni vanno anteposte ad altre ed hanno queste funzioni:

B - esegue la macro seguente con la matita sollevata, senza tracciare linee.

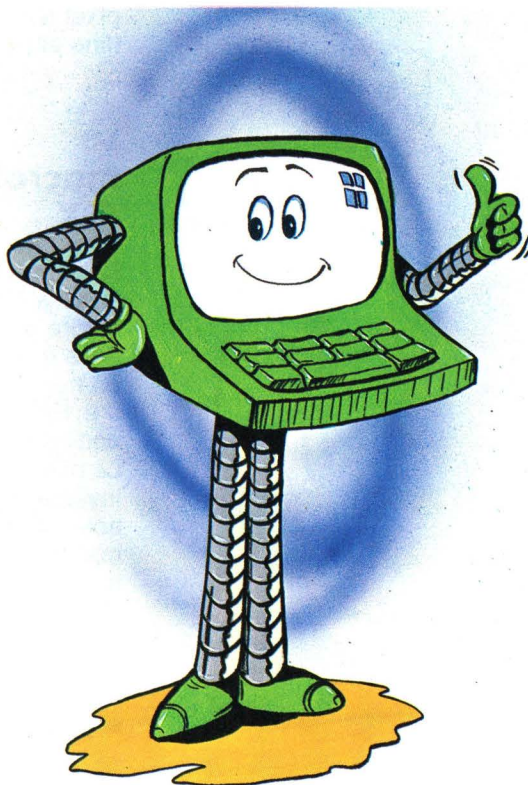
N - dopo l'esecuzione della macro successiva, torna al punto di partenza.

Esempi

```
DRAW "BR15"
```

Sposta la matita di 15 pixel a destra senza tracciare linee.

```
DRAW "NE20NF20NG20NH20"
```

traccia una grande X, riportando la matita al centro.

La macroistruzione M

Con la macroistruzione M è possibile spostare la matita in qualsiasi punto dello schermo, semplicemente fornendone le coordinate.
Numeri senza segno indicano coordinate assolute; numeri con segno indicano invece coordinate relative.
Naturalmente anche questa macroistruzione può essere preceduta dalla lettera B o N con i significativi visti prima.

Esempio

```
DRAW "M10,20BM+5,-3"
```

Traccia una linea fino al punto di coordinate 10,20; poi spostati 5 pixel a destra e 3 in alto (fino al punto 15,17).

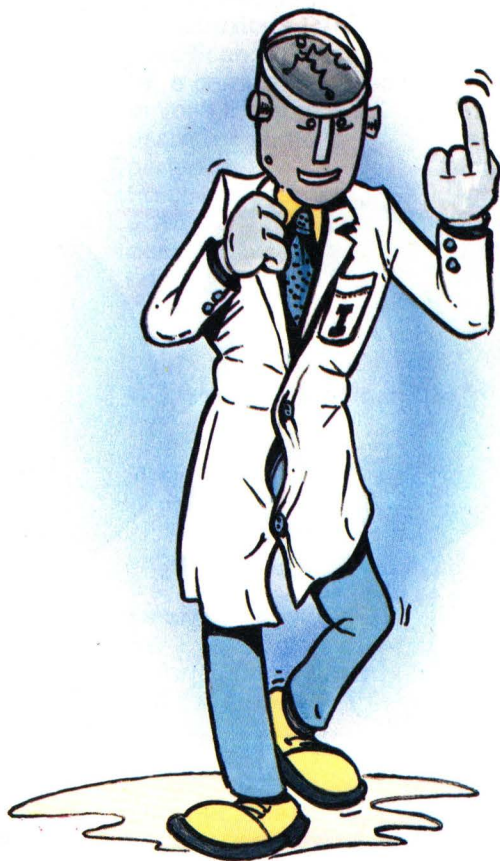
Le macroistruzioni C,X

La macroistruzione C serve a cambiare il colore della matita, e deve essere seguita da un numero compreso tra 0 e 15 indicante il codice del colore.
La macroistruzione X, invece, serve ad eseguire le macroistruzioni contenute in una variabile stringa.
Per fare questo basta scrivere, dopo la lettera X, il nome della variabile seguito dal carattere ";".

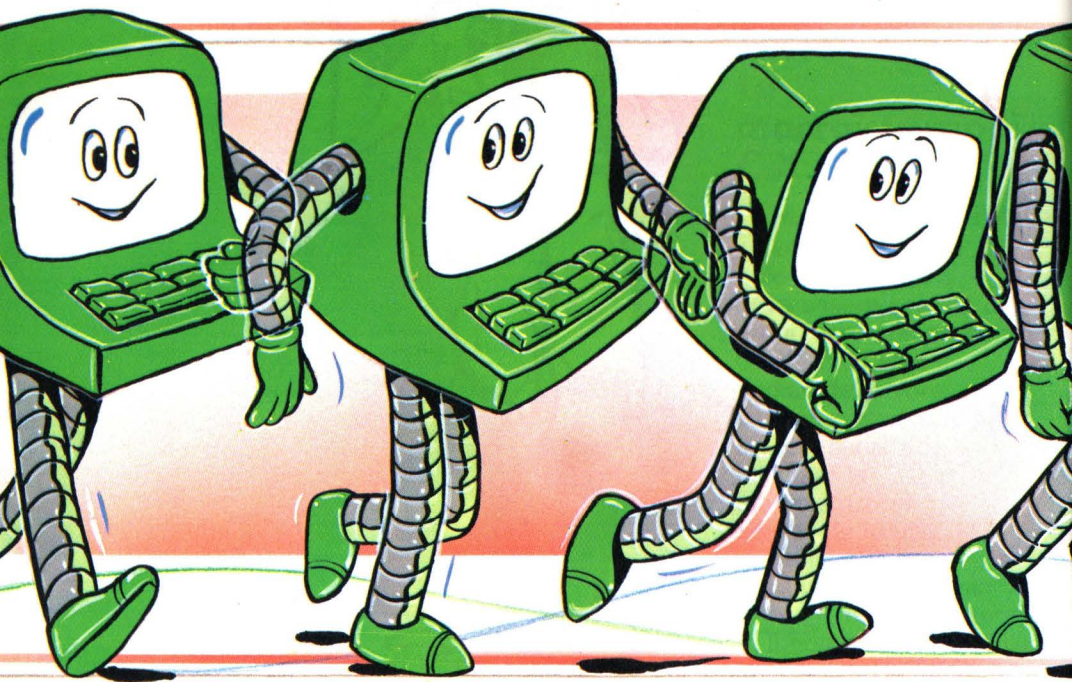
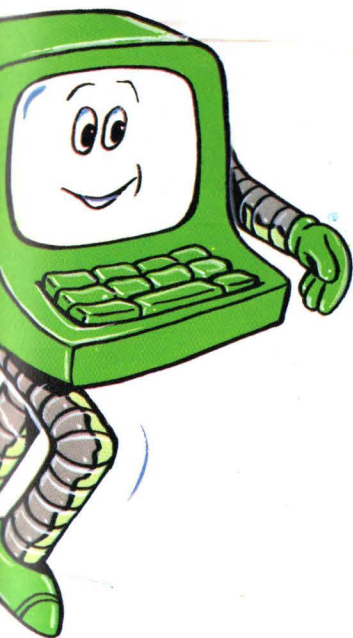
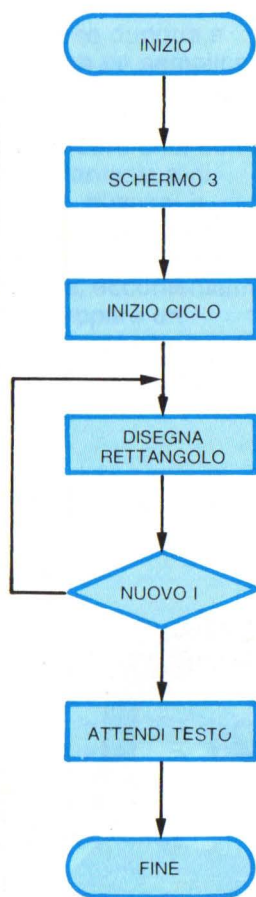
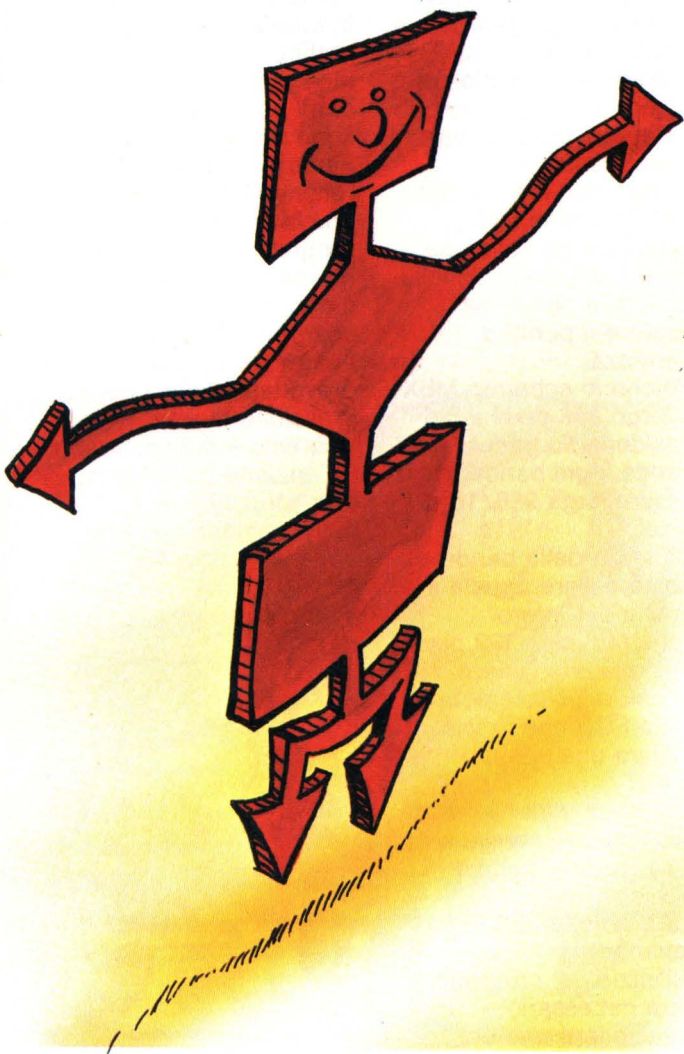
```
DRAW "C15R10C1D5"
```

Traccia una linea di 10 pixel verso destra usando il colore bianco, poi una linea di 5 pixel verso il basso usando il nero.

```
DRAW "BM10,10XDR$;"
```

Sposta la matita al punto di coordinate (10,10), e poi esegui le macroistruzioni contenute nella stringa DR$.

Programmazione grafica

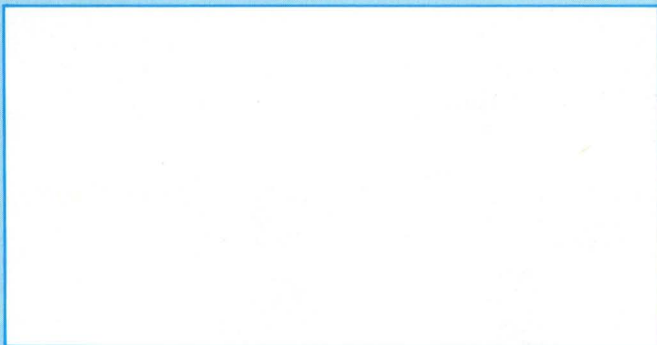
La grafica è senz'altro uno dei campi della programmazione che lascia più spazio alla fantasia e alla creatività. Il BASIC MSX offre istruzioni molto potenti nel campo della grafica, e rende possibili applicazioni molto interessanti facendo uso di programmi relativamente semplici.

Proviamo dunque a costruire un semplice programma che usa le istruzioni grafiche, ma sono sicuro che, con un po' di fantasia e di pratica, riuscirai a realizzare programmi molto più sofisticati. Per ora, accontentiamoci di sviluppare un programma che tracci sullo schermo delle

bande verticali rettangolari con tutti i colori MSX.
Un simile programma può esserti utile per mettere a punto le regolazioni del tuo televisore, o per conoscere meglio gli effetti derivanti dall'accoppiamento dei colori.
Avrai già intuito che per tracciare le bande rettangolari faremo uso dell'istruzione LINE specificando l'opzione, BF e cambiando di volta in volta il colore da usare e il punto di partenza.
Poiché lo schermo MSX è largo 256 pixel e noi desideriamo tracciare 16 bande, ogni banda dovrà essere larga 256/16=16 pixel.
L'altezza delle bande dovrà essere uguale a quella dell'intero schermo, cioè 192 pixel.
Per ottenere l'intero set di colori, utilizzeremo un ciclo FOR che faccia variare una variabile da 0 a 15.
La stessa variabile usata per il colore servirà a determinare il punto in cui iniziare il tracciamento del rettangolo.
All'inizio del programma sarà necessario selezionare uno dei modi grafici 2 o 3.
Il fatto che le coordinate grafiche siano calcolate sulla stessa base sia in modo 2 che in modo 3 consente di utilizzare indifferentemente il primo o il secondo.
In modo 3, però, la grafica è gestita con maggiore velocità, dato che i pixel da controllare sono in numero minore.
Finito il tracciamento dei rettangoli, per evitare che la fine del programma faccia ritornare lo schermo in modo testo facendo sparire il frutto del nostro lavoro, dovremo prevedere l'attesa della pressione di una tasto.
Useremo a questo scopo la funzione INPUT$, in una istruzione di assegnazione con una variabile fittizia A$.

```
10 SCREEN 3
20 FORI=0 TO 15
30 LINE (I*16,0) - STEP(16,192), I, BF
40 NEXT I
50 A$ = INPUT$ (1)
```

PROGRAMMAZIONE

L'errore è sempre in attesa: malignamente aspetta una tua distrazione, anche minima.
Non dargli mai la possibilità di sogghignare.
Impara la sintassi delle istruzioni.
Realizza programmi in forma modulare, facilmente leggibili, e molto documentati.
Nella digitazione, poi, metti la massima attenzione: anche una virgola al posto sbagliato, compromette la buona riuscita del programma.

VIDEOESERCIZI

Le enormi capacità grafiche del tuo MSX mettono a tua disposizione uno strumento creativo davvero straordinario. Sulla base di questi esempi applica la tua fantasia e cerca nuovi effetti.
Il successo presso amici e parenti è garantito.

```
10 REM QUADRANTI
20 SCREEN 2
30 DRAW "BM128,96"
40 FOR I=4 TO 255 STEP 3
50 XC=I MOD 13+2
60 XA=(I/4) MOD 3
70 DRAW"S=I;C=XC;A=XA;URDL2UR2DL"
80 NEXT I
90 GOTO 90
```

```
10 REM STELLA DI LINEE
20 SCREEN 2
30 FOR T=0 TO 256 STEP 4
40 LINE (T,0)-(256-T,192)
50 LINE (0,T)-(255,192-T)
60 NEXT T
70 GOTO 70
```